Microsoft

.net developer training

Create XML Web Services and Applications with Visual Studio .NET

Configuring and Deploying .NET Applications

Module 13 Agenda

- .NET Framework Provides for Application Deployment
- Packaging Applications
- Assemblies
- .NET Avoids DLL Hell
- Configuration Files
- Deployment
- Installation Services
- Maintenance

.NET Framework Provides For Application Deployment

- **Applications are easy to deploy in .NET**
  - **No-impact applications (application isolation)**
  - **Private components by default**
  - **Controlled code sharing**
  - **Side-by-side versioning**
  - **XCOPY deployment and replication**

.NET Framework Provides For Application Deployment

- Applications are easy to deploy
  - On-the-fly updates
  - Windows installer
  - Enterprise deployment
  - Download and caching
  - Partially trusted code

Packaging Applications

- Options
  - Single assembly
  - .cab files
  - Windows installer package

Assemblies
Private and Shared

- **Assemblies**
  - Is the deployment unit
  - Security boundary
  - Type boundary
  - Version boundary
  - Side-by-side execution

Assemblies
Private Assemblies

- Only visible for the application
- Stored in application folder or subdirectory
- Isolated from changes to the system
- No naming requirements (except uniqueness to application)
- Always loaded from the application directory – never from the GAC
  - The GAC requires both a strong name and a version

Assemblies
Shared Assemblies

- Assembly shared among several applications
- More stringent naming requirements (Strong names)
  - Uniqueness
  - Protection of the name
- Typically deployed to GAC
  - May also be “privatized” to application directory
- Versions are checked and version policies apply

.NET Avoids DLL Hell

- No registration of components
- No copying to %windir%\System32 needed
- Side-by-side versioning for shared components
  - For components in GAC
- No risk of overwriting components

Configuration Files
Machine, Application, Web

- Several text files control application configuration
  - ASP .NET reads all relative configuration files
    - Machine – MyApp.exe.Config
    - Application - Application.config
    - Parent/Sub/ThisFolder
    - Security - Web.Config
    - Combines settings from all files
    - Complete set of settings controls application

Configuration Files
Machine.Config

- Controls global settings for server
  - Combines with application settings
    - Specific settings in Machine.Config override application settings

Configuration Files
Application

- Specific to executable applications (.exe)
  - File named similar to application
    - MyApp.exe
    - Config is: MyApp.exe.Config
    - Exists in the same folder as executable application

Configuration Files
Web.Config

- Configuration file for ASP .NET applications
- Multiple files are allowed
  - One per folder
    - Each Web.Config file controls setting for that folder and all sub folders
    - Multiple Web.Config files are combined
      - MyApp - Web.Config
        SubApp - Web.Config
        MiniApp - Web.Config

Configuration Files
Web.Config

- XML configuration file is application scoped
- Deploys with the application

```
<location path = “Products.aspx">
        <security>
            <authorization>
                <deny users = "?" />
            </authorization>
        </security>
    </location>
```

Configuration
Demo

Machine.Config

machine.config - Notepad

File Edit Format Help Send

```
<?xml version="1.0" encoding="UTF-8" ?>

<configuration>

    <configSections>
        <!-- tell .NET Framework to ignore CLR sections -->
        <section name="runtime"  type="System.Configuration.IgnoreSectionHandler, System, Version=1.0.2411.0, Culture=neutral, PublicKeyTok
        <section name="mscorlib" type="System.Configuration.IgnoreSectionHandler, System, Version=1.0.2411.0, Culture=neutral, PublicKeyTok
        <section name="startup"  type="System.Configuration.IgnoreSectionHandler, System, Version=1.0.2411.0, Culture=neutral, PublicKeyTok
        <section name="system.runtime.remoting" type="System.Configuration.IgnoreSectionHandler, System, Version=1.0.2411.0, Culture=neutra

        <section name="system.diagnostics" type="System.Diagnostics.DiagnosticsConfigurationHandler, System, Version=1.0.2411.0, Culture=ne
        <section name="appSettings" type="System.Configuration.NameValueFileSectionHandler, System, Version=1.0.2411.0, Culture=neutral, Pu

        <sectionGroup name="system.net">
            <section name="authenticationModules" type="System.Net.Configuration.NetAuthenticationModuleHandler, System, Version=1.0.2411.0
            <section name="defaultProxy" type="System.Net.Configuration.DefaultProxyHandler, System, Version=1.0.2411.0, Culture=neutral, P
            <section name="connectionManagement" type="System.Net.Configuration.ConnectionManagementHandler, System, Version=1.0.2411.0, Cu
            <section name="webRequestModules" type="System.Net.Configuration.WebRequestModuleHandler, System, Version=1.0.2411.0, Culture=n
        </sectionGroup>

        <section name="system.windows.forms" type="System.Windows.Forms.WindowsFormsSectionHandler, System.Windows.Forms, Version=1.0.2411.

        <sectionGroup name="system.web">
            <section name="browserCaps" type="System.Web.Configuration.HttpCapabilitiesSectionHandler, System.Web, Version=1.0.2411.0, Cult
            <section name="clientTarget" type="System.Web.Configuration.ClientTargetSectionHandler, System.Web, Version=1.0.2411.0, Culture
            <section name="compilation" type="System.Web.UI.CompilationConfigurationHandler, System.Web, Version=1.0.2411.0, Culture=neutra
            <section name="pages" type="System.Web.UI.PagesConfigurationHandler, System.Web, Version=1.0.2411.0, Culture=neutral, PublicKey
            <section name="customErrors" type="System.Web.Configuration.CustomErrorsConfigHandler, System.Web, Version=1.0.2411.0, Culture=
            <section name="httpRuntime" type="System.Web.Configuration.HttpRuntimeConfigurationHandler, System.Web, Version=1.0.2411.0, Cul
            <section name="globalization" type="System.Web.Configuration.GlobalizationConfigurationHandler, System.Web, Version=1.0.2411.0,
            <section name="httpHandlers" type="System.Web.Configuration.HttpHandlersSectionHandler, System.Web, Version=1.0.2411.0, Culture
            <section name="httpModules" type="System.Web.Configuration.HttpModulesConfigurationHandler, System.Web, Version=1.0.2411.0, Cul
            <section name="iisFilter" type="System.Web.Configuration.IisFilterConfigurationHandler, System.Web, Version=1.0.2411.0, Culture
            <section name="processModel" type="System.Web.Configuration.ProcessModelConfigurationHandler, System.Web, Version=1.0.2411.0, C
            <!-- security -->
            <section name="identity" type="System.Web.Configuration.IdentityConfigHandler, System.Web, Version=1.0.2411.0, Culture=neutral,
            <section name="authorization" type="System.Web.Configuration.AuthorizationConfigHandler, System.Web, Version=1.0.2411.0, Cultur
            <section name="authentication" type="System.Web.Configuration.AuthenticationConfigHandler, System.Web, Version=1.0.2411.0, Cult
            <section name="machineKey" type="System.Web.Configuration.MachineKeyConfigHandler, System.Web, Version=1.0.2411.0, Culture=neut
            <!-- /security -->
            <section name="sessionState" type="System.Web.SessionState.SessionStateSectionHandler, System.Web, Version=1.0.2411.0, Culture=
            <section name="trace" type="System.Web.Configuration.TraceConfigurationHandler, System.Web, Version=1.0.2411.0, Culture=neutral
            <section name="trust" type="System.Web.Security.CodeAccessSecurityHandler, System.Web, Version=1.0.2411.0, Culture=neutral, Pub
            <section name="securityPolicy" type="System.Web.Configuration.SecurityPolicyConfigHandler, System.Web, Version=1.0.2411.0, Cult
            <section name="webControls" type="System.Configuration.SingleTagSectionHandler, System, Version=1.0.2411.0, Culture=neutral, Pu
            <section name="webServices" type="System.Web.Services.Configuration.WebServicesConfigurationSectionHandler, System.Web.Services
        </sectionGroup>
    </configSections>

    <appSettings>
        <!-- use this section to add application specific configuration
             example: <add key="XML File Name" value="myXmlFileName.xml" /> -->
    </appSettings>
```

Deployment
Windows Application

- No-nonsense - Just XCOPY or FTP
- Visual Studio® Deployment project
  - CAB files and MSI packages
- Auto-deployment for Windows® Applications
  - Windows Forms and controls can auto download over HTTP

Deployment
Web Application

- Deploy by XCOPY to server
  - Works for WebPages, WebServices, Compiled Components and Configuration (Meta) Data
- Web copy project from Visual Studio® .NET
  - Uses FrontPage® server extensions over HTTP
- Advantages:
  - No registration needed
  - No local server access needed
  - No Web server restarts needed

Deployment
XCOPY

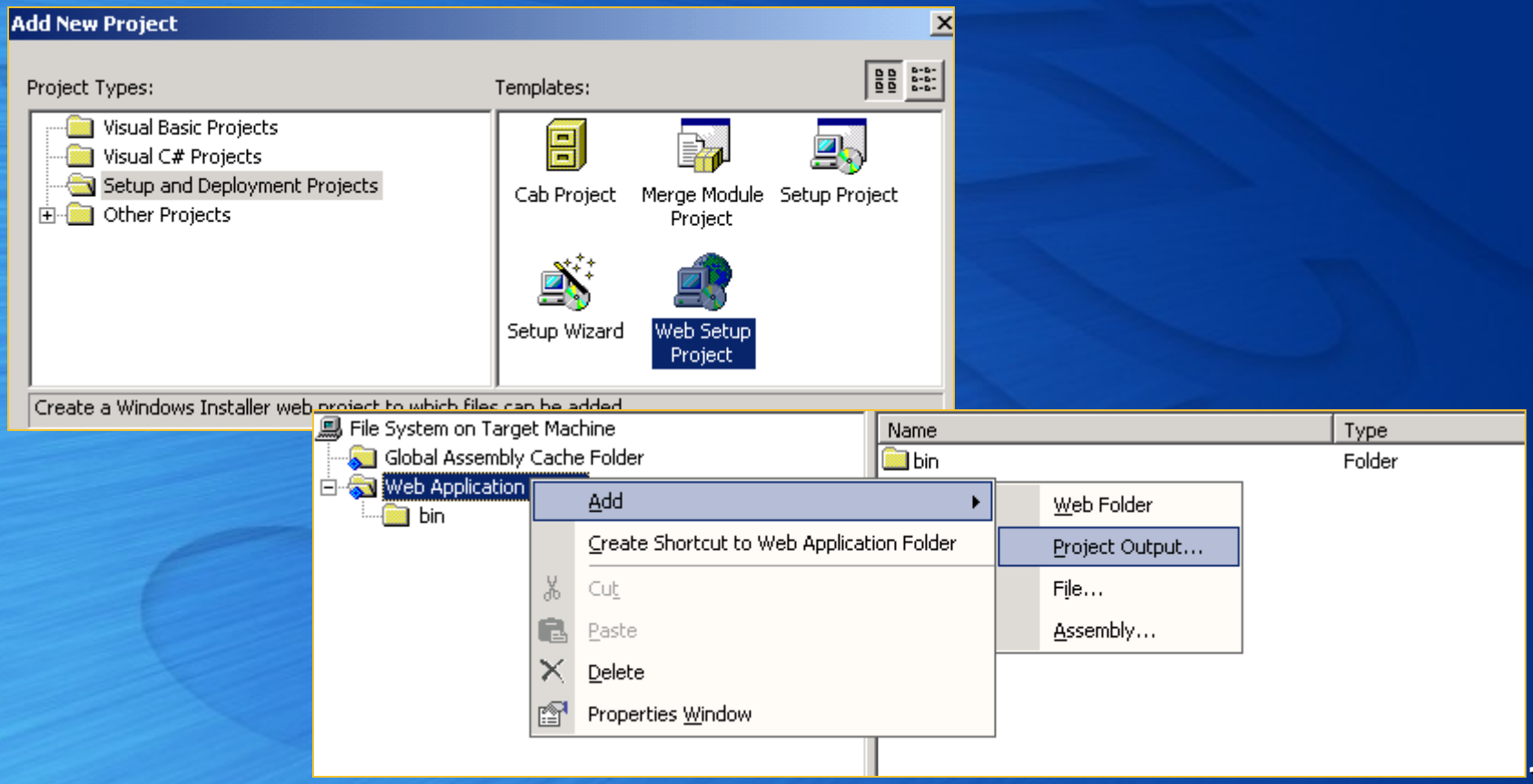
- Copy assemblies to application directory
- Local or network path allowed
- .NET CLR required on executing machine
- Uninstall means deleting assemblies

Global Assembly Cache

- Machine-wide code cache
- Stores assemblies specifically designated to be shared by several applications
- To deploy an assembly into the GAC
  - Use an installer designed to work with the GAC
    - Windows installer correctly installs
      - Handles reference checking
- Local copy settings control where assembly exists

Configure And Deploy
Demo

- **Configuration and Deployment Demo**
  - **Can use XCOPY, FTP, Visual Studio .NET**



Deployment Using Installation Services

- Windows Installer 2.0 supports assemblies
  - Use assemblies as components
  - All Windows Installer features available:
    - Resiliency (recovers from broken installations)
    - On-demand installation
    - Maintenance installation
    - Application rollback
- Supported by
  - SMS and IntelliMirror®
  - Automated deployment - Zero Administration for Windows (ZAW)

Custom Installers Deployment

- **Custom installers**
  - **Set up resources during deployment (e.g., create a message queue)**
  - **Run user code at install time (at the end of setup)**
- **Dynamic properties**
  - **Useful for Web services**
  - **Property values can be changed after the project is built and deployed**
  - **RAD designer support to “bind” to configuration settings**

Installation Services
Demo

- Installation Services Demo
- Use Installer to create an install package; use MSI setup

Maintenance

- Installing new version of private assembly
  - Replace original file
  - Developer is responsible for function
  - Version checking and policies don’t apply
- Installing QuickFix of assembly
  - Applications automatically use QuickFix
- Install new version of shared assembly
  - Install in GAC (or other loc)
  - Depending on version policy (publisher, application or administration), application uses new or previous version

Maintenance
Demo

- Maintenance Demo
- Change and compile new version of assembly
  - Install assembly on server while application is working

Summary

- Configuration
  - Uses text files
  - XML format makes it easy to configure
  - Many options for granular control over application
- Deployment is simpler
  - No registration
  - Easy and safe replacement of components

Resources

- ***Basics of Distributed Applications*** **- Multiple links about the pieces that make up a distributed application http://www.gotdotnet.com/team/clr/about_distributedapps.aspx**
- ***Deploying ASP .NET Applications*** **- A look at how to deploy .NET Applications http://www.fmexpense.com/quickstart/aspplus/doc/deployment.aspx**

Microsoft®

13.29